HIGH LEVEL GAME DESIGN PRODUCTION PROCESS SEQUENCE **Sequence of Game Development Stages**

1. **High Concept Document**
   * **Description**: A concise summary outlining the core idea and vision of the game; the barest form of the concept. This needs to be made of simple categories flowing from the idea collection system.
2. **Pitch Document**
   * **Description**: Combines the High Concept with initial concept art to present the game idea to stakeholders or potential investors. This is where concept art is added as well.
3. **Concept Document**
   * **Description**: An expanded document including preliminary market analysis, feasibility studies, and detailed concept explanations. Consider defining a game concept by listing the components ie systems of mechanics.
4. **Paper Prototype**
   * **Description**: A physical model representing the core gameplay loops; requires accompanying documentation to fully understand the game mechanics.
5. **Game Design Document (GDD)**
   * **Description**: A comprehensive document detailing gameplay mechanics, story elements, features, and overall design plans. Consider using Unity ECS/DOTs structure to outline GDD.
6. **Rapid Prototype**
   * **Description**: An initial digital version testing core mechanics using placeholder assets; still heavily reliant on the GDD for context.
7. **Blockout Prototype**
   * **Description**: A more refined digital prototype with basic level layouts using simple shapes or placeholder graphics (also known as "blockmesh" in 2D games).
8. **Internal Prototype (First Playable)**
   * **Description**: A version integrating core mechanics and basic visuals, allowing internal teams to experience early gameplay; starts to stand on its own but still benefits from documentation.
9. **Vertical Slice**
   * **Description**: A fully playable section showcasing final quality in a limited scope; represents the game's vision and can be understood without additional documentation.
10. **Alpha Build**
    * **Description**: A feature-complete version covering the entire game, but with unpolished content and placeholder assets in some areas; playable from start to finish.
11. **Beta Build**
    * **Description**: A content-complete and polished version focusing on bug fixes, stability, and balance; ready for external testing with minimal need for documentation.
12. **Release Candidate**
    * **Description**: A near-final version intended for release pending final testing; all content and features are locked, and the game is self-explanatory.
13. **Official Version (Gold Master)**
    * **Description**: The final, approved version released to the public; the game fully represents the intended product.
14. **Post-Launch Patch (Day-One Patch)**
    * **Description**: Updates released immediately after launch to fix minor issues, improve performance, and make last-minute adjustments.
15. **Content Update**
    * **Description**: Free additions of new content or features to keep players engaged, such as new items, events, or game modes.
16. **Paid Expansion Edition #**
    * **Description**: Major, premium updates adding substantial new content like storylines, areas, or gameplay systems; extends the game's life.
17. **Complete Edition**
    * **Description**: A package including the base game and all released expansions and updates; offers the full experience in one product.
18. **Remastered Edition**
    * **Description**: An enhanced version with improved graphics, performance, and possibly additional quality-of-life improvements; updated for newer hardware.
19. **Definitive Edition**
    * **Description**: The most complete and enriched version, including all content from the remastered edition plus exclusive additions or enhancements.
20. **Legacy Edition**
    * **Description**: A preservation-focused version ensuring continued playability and compatibility with future technology; minimal gameplay changes, emphasizing long-term access.
21. **Open Source Edition**
    * **Description**: The game's source code and assets are made publicly available under a specific license, allowing the community to modify, preserve, and build upon the game.

**Key Considerations and Notes**

* **Product Progression Focus**: The sequence emphasizes the progression of the game product itself at each stage, rather than the processes or activities that advance the product.
* **Dual Tracks of Development**:
  + **Documentation**: Essential in early stages for understanding the game's vision, mechanics, and design decisions.
  + **Digital Product**: Becomes increasingly self-explanatory as development progresses, reducing reliance on documentation.
* **Granularity and Industry Terms**: The sequence aims to be as granular as possible, using industry-recognized terms to describe each advancement, even if some stages are squeezed into places they might not traditionally fit.
* **Understanding Grayboxing in 2D**:
  + In 2D game development, grayboxing is often referred to as "blockmesh" or "blockout" prototyping.
  + It involves using simple shapes or placeholder graphics to outline level layouts and gameplay spaces.
* **Rapid Prototyping vs. Grayboxing**:
  + **Rapid Prototyping**: An initial, rough attempt to validate core game mechanics; simpler and less detailed.
  + **Grayboxing**: A step further, focusing on level design and spatial arrangements with basic visuals.
* **Progression of Prototypes**:
  + Each prototype builds upon the previous, adding layers of complexity, refinement, and polish.
  + The aim is to progressively reduce the need for external documentation by making the game itself convey all necessary information.

**Suggestions for Future Development**

* **Stakeholder Roles and Contributions**:
  + Explore how different team members (developers, designers, QA testers, marketing teams, player community) contribute at each stage.
  + Understand how responsibilities and involvement shift as the game progresses.
* **Community Engagement**:
  + Delve into the increasing role of the player community in post-launch stages.
  + Consider how community feedback, modding, and open-source contributions impact the game's evolution.
* **Workflow and Handoffs**:
  + Map out the collaborative processes between different departments during development.
  + Examine how handoffs and interdependencies affect the progression and quality of the game.
* **Detailed Documentation and Deliverables**:
  + Identify specific documents and deliverables produced at each stage (e.g., updated GDDs, technical design documents, patch notes).
  + Assess how these support development, communication, and alignment within the team.

**Important Points from Our Conversation**

* **Desire for Granularity**: You emphasized wanting the most granular and in-sequence progression imaginable, even if it means including steps that might not traditionally be recognized in that order.
* **Focus on the State of the Product**: You requested to trim the sequence to just the things that describe the state of the product itself, not the processes or activities that advance it.
* **Two Tracks (Documentation and Digital Product)**: Recognizing that in many stages, both the documentation and the digital product are necessary to fully understand and represent the game.
* **Assistant's Role**: The assistant provided detailed explanations of each stage and suggested exploring stakeholder roles as a next logical step for further development.

**Next Steps and Prompt for Future Conversations**

To continue building upon this foundation, consider the following prompt:

"Based on this sequence of game development stages, analyze how different stakeholders (developers, designers, QA testers, marketing teams, and the player community) contribute at each step. Describe how their responsibilities evolve as the game progresses from concept to open source, and explore the collaborative processes that drive each stage forward."

This summary captures all the relevant information from our previous discussion, ensuring that you can quickly get back up to speed in future conversations or projects. It includes the sequence of stages, key considerations, and suggestions for further exploration, providing a solid foundation for continued analysis of game development processes.